Kent
Business
School

ISSN 1748-7595 (Online)

## Working Paper Series

# On Centralised Resource Allocation Using DEA

Cecilio Mar-Molinero
Kent Business School

Diego Prior
Department of Business Economics
Universitat Autonoma de Barcelona

KBS
UNIVERSITY OF KENT

**Working Paper No.121**
*March 2006*

**On centralised resource allocation using *DEA***

Cecilio Mar-Molinero (*)
Kent Business School
University of Kent, UK.
c.mar-molinero@kent.ac.uk

Diego Prior
Dpt. of Business Economics
Universitat Autònoma de Barcelona
Barcelona, Spain.
diego.prior@uab.es

This version March 2006

(*): Corresponding author

**Abstract**

The standard *DEA* model allows different *DMU* units to set their own priorities for the inputs and outputs that form part of the efficiency assessment. In the case of a centralised organisation with many outlets, such as an education authority that is responsible for many schools, it may be more sensible to operate in the most efficient way but under a common set of priorities for all the *DMUs*. The algorithm that is used to do this, the centralised resource allocation model, does just this. We show that the centralised resource allocation model can be substantially simplified. We interpret the simplifications and show how the model works using a standard data. It is shown that the most desirable *DMU* is found as a by-product of the estimation. This is useful information when planning new units.

Keywords: *DEA*, centralised planning, efficiency.

## 1. Introduction

Under the standard *DEA* model, each Decision Making Unit (*DMU*) sets its own priorities for inputs and outputs. This is reasonable when each *DMU* operates independently, but does not make much sense when the *DMUs* are under the central control of a decision maker who would like to see a common set of priorities operating over all the system. Examples of centralised decision making where all DMUs would be expected to behave in the same way are the branches of a bank; the schools in a local authority; and the organisation of a local service such as refuse collection under the responsibility of a common council. Take, for example, schools, one may just ask why a teacher should be valued differently in two different schools when doing the same job, in the same way, for the same education authority. It would be much more reasonable impose the same model on all the *DMUs* under the same decision maker. This is exactly what Lozano and Villa (2004) proposed.

In this paper we interpret the Lozano and Villa (2004) model, propose simplifications and extensions, and reinterpret what they define as variable returns to scale (*VRS*).

This introduction is the first section of the paper. The second part is concerned with the model and its properties. The paper continues with an example using data from the literature, and ends with a conclusion.

## 2. The model

Lozano and Villa (2004) suggest a variety of models but here we will only discuss one of them: *"Model Phase I/Radial/ Input-Oriented"*. This model can be formulated in the envelopment form and in the ratio form. In order to make the discussion easier to follow we will reproduce both versions here, starting with the ratio formulation. We will preserve the notation of the original paper.

The formulation of *"Model Phase I/Radial/ Input-Oriented"* is as follows:

$$
\begin{aligned}
&\max \frac{\sum_{k=1}^{p} n_k \sum_{r=1}^{n} y_{kr} + \sum_{r=1}^{n} x_r}{\sum_{i=1}^{m} u_i \sum_{j=1}^{n} x_{ij}} \\
&s.t. \qquad\qquad\qquad\qquad\qquad\qquad [1] \\
&\frac{\sum_{k=1}^{p} n_k y_{kj} + x_r}{\sum_{i=1}^{m} u_i x_{ij}} \leq 1, \qquad \forall j, \forall r \\
&u_i \geq 0, \qquad n_k \geq 0, \qquad x_r \ \mathit{free}
\end{aligned}
$$

Where $?_k$ is the weight associated with output $k$, of which there are $p$,
$u_i$ is the weight associated with input $i$, of which there are $m$,
$y_{kr}$ is the quantity of output $k$ generated by *DMU r*,
$x_{ij}$ is the quantity of input $i$ that is used by *DMU j*,
$?_r$ is a *VRS* variable associated with *DMU r*,
there are *n DMUs* in the system.

In this model, the objective function values all the outputs, irrespective of the *DMU* that generates them, at the same price, $?_k$. In this way, the numerator of the fraction gives the total valuation of the outputs of the system while the denominator of the fraction performs the same function with inputs, each input being weighted by $u_i$, irrespective of the *DMU* that uses it. In this sense, the whole organisation is being treated as a giant *DMU* that uses all the inputs available in all the *DMUs* to generate all the outputs that the system generates irrespective of the *DMU* in which they are produced.

The constraints are the usual ones in *DEA*. For each *DMU*, the inputs that are used are valued at the overall price, $u_i$, and the outputs that are produced are also valued at the overall price, $?_k$. Note that each *DMU* has two subindexes, reflecting the fact that cross-efficiencies are computed under the returns to scale associated with every *DMU* in the system. We will show that the model can be simplified.

This model cannot be interpreted in the usual *DEA* way. We are no longer asking every *DMU* to choose the weights that make it look in the best light under the constraint that the remaining *DMUs* should be assessed with the same weights. There is a subtle change: the system, as a global unit, finds the weights that present it in the best light and assesses the performance of individual *DMUs* under these weights.

The equations for the envelopment formulation- also given by Lozano and Villa (2004)- are as follows.

$$
\begin{aligned}
&\min q \\
&s.t. \\
&\sum_{r=1}^{n}\sum_{j=1}^{n} l_{jr} x_{ij} \leq q \sum_{j=1}^{n} x_{ij}, \qquad \forall i \\
&\sum_{r=1}^{n}\sum_{j=1}^{n} l_{jr} y_{kj} \geq \sum_{r=1}^{n} y_{kr}, \qquad \forall k \\
&\sum_{j=1}^{n} l_{jr} = 1, \qquad \forall r \\
&l_{jr} \geq 0, \qquad q \ free
\end{aligned}
\qquad [2]
$$

This is not the standard *BCC* model, since there are two summatory signs in the input and in the output constraints. In fact, this model can easily be obtained from the *BCC* formulation. Under the standard *BCC* approach to *DEA* it is necessary to run the model for each *DMU*. The Lozano and Villa (2004) formulation just adds up all the individual *BCC* equations, for all the *DMU* runs, assuming a common value for *?*. The end result is that the right hand side of the input constraints contains all the inputs available to the system; and that the right hand side of the output constraints contains all the outputs that it has produced. A further consequence is that there is a *VRS* constraint for each *DMU*.

It is worth noting that the number of unknowns in this formulation- excluding slack variables- is $n^2+1$, as each *DMU*- of which there are *n*- creates *n ?s*, and the overall efficiency, *?*, is also an unknown. The number of unknowns to be estimated increases as a quadratic function of the number of *DMUs*. This has the consequence that problems with a relatively small number of *DMUs* become large quite quickly.

There is an interpretation for the above equations. The output constraints indicate that we would like to obtain at least the total amount of outputs that are currently being obtained from the system. The input constraints can be read as if we would be prepared to radially reduce inputs in order to achieve the amount of outputs already available. This reduction would be done keeping the proportion in which the inputs are used. The returns to scale constraints are attempting to keep the size of *DMUs* within the observed range of values.

Two questions will be addressed. The first question is how the model can be best interpreted in logical terms. The second question is how the model can be simplified and generalised.

We will start with the second question. Are there any valid simplifications to this model?

### 2.1. Simplifications

Take the constraints in the ratio form of the model *[1]*. These can be rewritten as follows:

$$\sum_{k=1}^{p} n_k y_{kj} - \sum_{i=1}^{m} u_i x_{ij} + x_r \le 0, \qquad \forall j, \forall r$$

In this equation, $y_{kj}$ and $x_{ij}$ are data and, therefore, fixed in advance. The optimisation procedure calculates $?_k$ and $u_i$. Thus, for a given *DMU*, $j$, we can define:

$$\sum_{k=1}^{p} n_k y_{kj} - \sum_{i=1}^{m} u_i x_{ij} = k_j$$

And the equation becomes:

$$k_j + x_r \leq 0, \qquad \forall j, \forall r$$

Hence, for every *DMU, j,* there could be up to *n* values *?r*. This would be perfectly compatible with the equations but, is it possible for each *DMU* to be associated with a variety of *?r*? Imagine that this was indeed the case, it would mean that a *DMU* can be operating at the same time under a variety of variable returns to scale, something that does not make sense. It follows that, for a given *j*, all the *?r* are equal. Since this happens for any value of *j*, it further follows that all the *?r* are equal. We conclude that there is a single value of *?r* , that we may simply call *?*. Thus, in the constraints of the ratio model the subindex *r* can be dropped. Doing this creates, for each *DMU, j, r* identical constraints. Only one such constraint is needed for each *DMU*, the remaining *r-1* constraints can be dropped from the formulation.

The objective function can also be simplified. It becomes:

$$\max \frac{\sum_{k=1}^{p} n_k \sum_{r=1}^{n} y_{kr} + nx}{\sum_{i=1}^{m} u_i \sum_{j=1}^{n} x_{ij}}$$

Duality theory tells us that each constraint in the dual is associated with a column in the primal. Saying that dual constraints are not necessary implies that the associated primal variables are not needed either. In fact, if we work backwards, the envelope form of the model can be simplified to:

$$
\begin{aligned}
&\min \ q \\
&s.t. \\
&\sum_{j=1}^{n} l_j x_{ij} \le q \sum_{j=1}^{n} x_{ij}, \qquad \forall i \\
&\sum_{j=1}^{n} l_j y_{kj} \ge \sum_{j=1}^{n} y_j, \qquad \forall k \\
&\sum_{j=1}^{n} l_j = n \\
&l_j \ge 0, \qquad q \ \ free
\end{aligned}
\qquad [3]
$$

This formulation only contains *n+1* unknown decision variables, the *?j* and *?*. This is an important simplification with respect to the Lozano and Villa (2004) formulation. We will now proceed to interpret the model.

## 2.2. Interpretation: cloning the best *DMU*

The simplified model has a very similar structure to the *BCC* model, but there are some changes. The similarities are obvious: the left hand side of the input constraints and the left hand side of the output constraints remain unchanged with respect to *BCC*; the left hand side of the *VRS* constraint is also unchanged; and the objective function is the same.

The differences with *BCC* appear on the right hand side of the constraints. The right hand side of the input constraints contains the total amount of inputs available to the system, as in the original Lozano and Villa (2004) model. The right hand side of the output constraints is also the same as in the Lozano and Villa (2004) model, containing the total amount of output produced by the system. The interpretation of the input and output constraints remain unchanged: the system as a whole would like to produce at least as much as the current level of outputs, and will do this by radially reducing the total amount of inputs used.

The constraint that replaces the standard constraint for *VRS* is very interesting. Imagine -the unlikely situation- that the system is already operating under optimal conditions and using common weights. In this case the best way to produce the outputs is the current one, implying that all the $?_j$ should be equal to one; the sum of the $?_j$ becomes automatically equal to the number of *DMUs* in the system. It is easy to conjecture what will happen if the system is not operating under optimal conditions: the most efficient *DMUs* will be identified and "cloned" a certain number of times. The relative proportions under which these *DMUs* are operating will, in general, not be the same as the relative proportions under which the system as a whole operates, and other *DMUs* will be used in order to make up the acceptable balance of inputs and outputs. This is, in fact, the interpretation given by Lozano and Villa (2004), although they do not discuss the issue of *VRS*.

The formulation has a further consequence: it may be possible to identify the most efficient *DMUs*, as these are the ones "cloned", but it is not possible to know for each *DMU* the returns to scale under which it is operating.

We should think further about modelling issues. Is it really necessary for the sum of the $?_j$ to be equal to the number of *DMUs*? This appears to be an unnecessarily tight constraint. One can indeed, think of situations when a solution to the problem could be found with a smaller number of *DMUs* than the current number. The way to model such situation is straight forward; all we have to do is to replace *n* on the right hand side of the lambda restriction of linear program *[3]* with a smaller number, very much as done by Lozano and Villa (2005). A further idea is that, we could be operating with more *DMUs* than we are currently operating, in order to have smaller decision units, but better balanced. It could even be desirable that we ought to consider a standard size of *DMU* that is reproduced as many times as necessary. An example of a situation where this would be desirable is school management: much debate has taken place on the optimal size of a school. The optimal size and balance of a school could be deduced from this model, and guidelines could be produced for the building of new schools, or for the reorganisation of existing schools.

We will now offer an example of the simplified formulation and we will compare it with the original formulation.

**3. Data and application**

In order to illustrate our proposal, we used a data set published by Zhu (1998) corresponding to 18 Chinese cities (13 open coastal and 5 special economic zones). Table 1 contains the values of the two inputs -investment in fixed assets by state owned enterprises, $x_1$, and foreign funds actually used, $x_2$- and the three outputs - total industrial output value, $y_1$, total value of retail sales, $y_2$, and handling capacity of coastal, $y_3$-. It is not our intention to investigate the efficiency of Chinese cities, but to demonstrate how the model works. Indeed, the results may not make any other than mathematical sense.

[Table 1 about here]

After running program *[2]*, we obtained the results shown in Table 2. The overall efficiency of the system is *0'7113*, meaning that it has been demonstrated that the outputs of the system could be produced while saving the *28'87%* of the inputs (*1-0'7113*).

[Table 2 about here]

When we run our simplified model, *[3]*, we obtain the same efficiency level, *0'7113*, while the *?s* the package returns replicate the last column of Table 2 (*S ?*) . It is worth discussing the meaning of this column. In order to globally minimise the inputs of the system, the most efficient *DMUs* are cloned. Yantai is cloned *10* times (in order to obtain a total of *11* units with similar characteristics); Dalian is cloned twice (and the system ends up with a total *3* identical units); Ningbo is cloned *1* time (to produce a total of *2* units); and, by defining *2* more units as a specific composite of Dalian, Yantai, Shanghai and Ningbo, the system saves the *28'87%* of the inputs while maintaining the total number *DMU's* in 18 (the same number we started with).

There are situations when the central planner can modify resource allocation by closing the most inefficient *DMU's* and/or by opening (cloning) new units. When this is the case, both linear programs *[2]* and *[3]* are artificially tight because they include a non-justifiable constraint (minimise inputs while maintaining the initial number of *DMU's*). In our particular case study we have conducted an experiment aimed at exploring the sensitivity of the solution to this constraint. To do this, we rerun program *[3]* by replacing $n$ with $\hat{n}$ - $\hat{n}$ taking values in the range $\hat{n} = (0.1)\,n$ to

$\hat{n} = (10)\,n$. The results of this exercise are displayed in Figure 1. There are no feasible solutions for $\hat{n} < 5$, meaning that it is impossible to produce the aggregated level of output with only *5 DMU's* or less. In the interval $5 < \hat{n} < 9$, there are feasible solutions, but the efficiency coefficient is bigger than the unity, meaning that this reallocation would consume more inputs than the initial situation, a result that is contrary to any notion of efficiency.

[Figure 1 about here]

Figure 1 shows that the solution presented in Table 2- $\hat{n} = 18$ and $q = 0.7113$ - is indeed a good solution; but it also shows that if we make $\hat{n} > 18$ we can obtain even better solutions. The overall minimum for $q$ (*0.35*) is reached when $\hat{n} = 50$ (cloning 43.75 times Weihai and 4.43 times Quinhuangdao). This solution can be easily obtained by dropping the constraint $\sum l$ from linear program *[3]*, which implies that that constants returns to scale is prevailing technology. It could be argued that Weihai is an ideal *DMU* and that, if new *DMUs* have to be built, an attempt should be made to take Weihai as an example of good practice.

## 5. Summary and conclusions

The original *DEA* model studied *DMUs* one at a time. Its philosophy was that each *DMU* wanted to be seen performing in the best possible way. Thus, each *DMU* was given the flexibility to value inputs and outputs in the way that best suited its "modus operandi".

Lozano and Villa (2004) made a significant contribution to the DEA literature by pointing out that this model was unrealistic when all the *DMUs* were under the control of a single decision maker. They proposed a formulation that valued inputs and outputs equally irrespective of the *DMU* that had used or produced them. In this paper we have attempted to interpret the Lozano and Villa (2004) model and we have shown that it can be substantially simplified. In the original model, the number of unknowns is proportional to the square of the number of *DMUs*, while in the simplified version the number of unknowns grows linearly with the number of *DMUs*. We think that this simplification makes the model easier to implement in many situations, for example, the number of schools under the control of a local authority may be rather large.

We have argued that the constraints that the sum of the lambdas should be equal to a certain number are unnecessarily restrictive. The link between this constraint and variable returns to scale is lost. This constraint does not serve to compare a *DMU* with a linear interpolation of existing *DMUs*, since we are no longer assessing individual *DMUs*. The only purpose of this constraint is to force the number of *DMUs* used in the final solution, both original and cloned, to a given total. This constraint can be totally relaxed if we want to discover the optimum size of a *DMU*, or can be given a value decided a priori if we want to limit the number of *DMUs* in the system.

We have given an example, based on data from the literature, to show that the original and the simplified models do indeed result in the same solutions.

## REFERENCES


Lozano, S. and G. Villa (2004), "Centralized Resource Allocation Using Data Envelopment Analysis", *Journal of Productivity Analysis*, 22, 143-61.

Lozano, S. and G. Villa (2005), "Centralized DEA models with the possibility of downsizing", *Journal of the Operational Research Society*, 56, 357-364.

Zhu, J. (1998), "Data envelopment analysis vs. principal component analysis: An illustrative study of economic performance of Chinese cities", *European Journal of Operational Research*, 111, 50-61.

**Table 1. Variables corresponding to Chinese open coastal cities and special economic zones (1989)**

| City | $y_1$ | $y_2$ | $y_3$ | $x_1$ | $x_2$ |
|---|---|---|---|---|---|
| Dalian | 160,89 | 80800,00 | 5092,00 | 2874,80 | 16738,00 |
| Qinhuangdao | 21,14 | 18172,00 | 6563,00 | 946,30 | 691,00 |
| Tianjin | 375,25 | 144530,00 | 2437,00 | 6854,00 | 43024,00 |
| Qingdao | 176,68 | 70318,00 | 3145,00 | 2305,10 | 10815,00 |
| Yantai | 102,12 | 55419,00 | 1225,00 | 1010,30 | 2099,00 |
| Weihai | 59,17 | 27422,00 | 246,00 | 282,30 | 757,00 |
| Shanghai | 1029,09 | 351390,00 | 14604,00 | 17478,60 | 116900,00 |
| Lianyunggang | 30,07 | 23550,00 | 1126,00 | 661,80 | 2024,00 |
| Ningbo | 160,58 | 59406,00 | 2230,00 | 1544,20 | 3218,00 |
| Wenzhou | 53,69 | 47504,00 | 430,00 | 428,40 | 574,00 |
| Guangzhou | 258,09 | 151356,00 | 4649,00 | 6228,10 | 29842,00 |
| Zhanjiang | 38,02 | 45336,00 | 1555,00 | 697,70 | 3394,00 |
| Beihai | 7,07 | 8236,00 | 121,00 | 106,40 | 367,00 |
| Shenzhen | 116,46 | 56135,00 | 956,00 | 4539,30 | 45809,00 |
| Zhuhai | 29,20 | 17554,00 | 231,00 | 957,80 | 16947,00 |
| Shantou | 65,36 | 62341,00 | 618,00 | 1209,20 | 15741,00 |
| Xiamen | 54,52 | 25203,00 | 513,00 | 972,40 | 23822,00 |
| Hainan | 25,24 | 40267,00 | 895,00 | 2192,00 | 10943,00 |

**Table 2. Results obtained from the application of program** ***[2] (? = 0,7113)***

| DMU number | | $?_1$ | $?_2$ | $?_3$ | $?_4$ | $?_5$ | $?_6$ | $?_7$ | $?_8$ | $?_9$ | $?_{10}$ | $?_{11}$ | $?_{12}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | **Dalian** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 2 | **Qinhuangdao** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | **Tianjin** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4 | **Qingdao** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 5 | **Yantai** | 1 | 1 | 0,76 | 1 | 0 | 1 | 1 | 0,84 | 0 | 1 | 0 | 1 |
| 6 | **Weihai** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 7 | **Shanghai** | 0 | 0 | 0,24 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 8 | **Lianyunggang** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 9 | **Ningbo** | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0,16 | 1 | 0 | 0 | 0 |
| 10 | **Wenzhou** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11 | **Guangzhou** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 12 | **Zhanjiang** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 13 | **Beihai** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 14 | **Shenzhen** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 15 | **Zhuhai** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 16 | **Shantou** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 17 | **Xiamen** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 18 | **Hainan** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | ***S?*** | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

**Figure 1. Evolution of the cloning process**